SYSTEM AND METHOD FOR RUNTIME REALLOCATION OF PLD RESOURCES

Neil G. Jacobson

FIELD OF THE INVENTION

The present invention generally relates to runtime reconfigurable systems, and more particularly to runtime reallocation of PLD resources in response to changing activity levels.

BACKGROUND

The field of reconfigurable computing has advanced steadily for the past decade, using programmable logic devices (PLDs) as the basis for high-performance reconfigurable systems. Run-Time Reconfigurable (RTR) systems distinguish themselves by performing circuit logic customization at run-time. RTR systems using PLDs are expected to result in systems that require less hardware, less software, and fewer input/output resources than traditional ASIC or PLD-based systems.

In multiprocessor data processing systems, various systems are known for balancing the work load between the processors. An advantage to such systems is that the functions implemented by the system can be software controlled, thereby providing great flexibility. However, such systems tend to be large, expensive, and complex. Furthermore, software-implemented functions tend to be slower than hardware implementations.

A system and method that address the aforementioned problems, as well as other related problems, are therefore desirable.

EL539650972US

SUMMARY OF THE INVENTION

A system and method for allocating resources of programmable logic devices (PLDs) according to the activity levels of the functions implemented on the PLDs are provided. A load monitor surveys the activity levels of the functions implemented on the PLDs. When decreasing and/or increasing activity levels are detected, the PLD resources are reallocated between the various functions in proportion to the decreasing and/or increasing activity levels.

In another embodiment, a reserve of PLD resources is maintained. When the activity level of one or more functions is increasing and none of the other functions have decreasing activity levels, some portion of the reserve is allocated to the functions having increasing activity levels.

Various other embodiments are set forth in the Detailed Description and Claims which follow.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a functional block diagram of a system 100 having control logic for reconfiguration of PLD resources and load monitoring of different functions that are implemented on one or more PLDs, in accordance with one embodiment of the invention.

FIG. 2 is a flowchart of a process for monitoring the activity level of the functions implemented on the PLD(s).

FIG. 3A is a flowchart of a process for determining the levels at which PLD resources should be reallocated between a number of functions when it is detected that the activity level for one of the functions is decreasing.

FIG. 3B is a flowchart of a process for determining the levels at which PLD resources should be allocated from the reserve of resources when the activity level of a function is increasing.

FIG. 4 is a flowchart of a process for controlling reconfiguration of PLDs in response to control parameters from the load monitor.

DETAILED DESCRIPTION

In various embodiments, the invention monitors the activity level of various functions that are implemented on one or more PLDs. When the activity level of a function has decreased, a determination is made as to whether PLD resources of the function should be reprogrammed for another function. If so, the PLD resources are reprogrammed accordingly.

FIG. 1 is a functional block diagram of a system 100 having control logic for reconfiguration of PLD resources and load monitoring of different functions that are implemented on one or more PLDs, in accordance with one embodiment of the invention. The system includes one or more PLDs 102 that implement different functions, a storage element 104 for storing one or more configuration bitstreams 106 that program the PLD(s) to implement the functions, a configuration control element 108, and a load monitor 110.

The invention is applicable to systems having any of a variety of PLDs, for example FPGAs or CPLDs, where the particular function and the resources available on the PLD will determine the number of PLDs required. Partial reconfigurability of the PLD will also influence the number PLDs in the system. For example, the Virtex FPGA from Xilinx supports partial reconfiguration. Therefore, multiple functions may be implemented on a Virtex FPGA, depending on the resource requirements of the different functions. If it is determined that FPGA resources should

be taken from one function and devoted to another function, the FPGA can be partially reconfigured for the reallocation of resources.

The different functions are implemented with associated configuration bitstreams. For example, configuration bitstream 106 implements a first function, configuration bitstream 112 implements a second function, configuration bitstream 114 implements a third function, and configuration bitstream 116 implements an *n*th function.

In one embodiment, storage 104 for the configuration bitstreams may include one or more in-system memory elements. In another embodiment, the configuration bitstreams may be stored outside the system on network storage, for example. Those skilled in the art will recognize other suitable storage alternatives for different application requirements.

Load monitor 110 is coupled to the PLDs for monitoring the activity level of the functions. For example, the PLDs may implement a plurality of communications channels, wherein different ones of the channels are dedicated to different protocols. If the traffic on the channels implementing a first protocol has decreased and the traffic on the channels implementing a second protocol has increased, some of the channels for the first protocol may be reallocated for the second protocol. The load monitor not only detects the need for reprogramming but also determines which PLD resources should be reprogrammed. Load monitor 110 also signals the functions implemented on the PLDs when a reconfiguration is to occur so that work-in-process can be finished or reassigned.

The mechanism by which load monitor 110 monitors the activity levels of the different functions can be adapted

in accordance with the requirements of the application. For example, the functions may be designed with a feature for reporting the activity levels via an output port of the PLDs. Alternatively, registers internal to the PLDs and readable by the load monitor may be used for storing the current activity levels.

In order to reconfigure PLD resources, load monitor 110 must track which functions are implemented on which PLD resources. Furthermore, if the different functions require different types and numbers of PLD resources, those too must be tracked by the load monitor.

Configuration control element 108 is coupled to load monitor 110, storage 104, and to the PLDs 102. When reprogramming is requested by the load monitor, configuration control 108 signals the selected PLD(s) and initiates transfer of the selected configuration bitstream(s) from storage 104 to the appropriate PLD(s). It will be appreciated that the logic of load monitor 110 and configuration control element 108 could be integrated into a single device in another embodiment. The particular distribution of functionality between load monitor 110 and configuration control element 108, as well as the particular communication links to PLDs 102, can be tailored for specific application requirements.

FIG. 2 is a flowchart of a process for monitoring the activity level of the functions implemented on the PLD(s). The process generally entails performing an initial allocation of PLD resources between the functions and an initial configuration of the PLDs and then monitoring activity levels of the functions and reconfiguring the PLDs as necessary.

At step 152, an initial allocation of PLD resources is selected for the various functions. For example, in an application such as a communications switch, certain numbers of channels may be dedicated to different communications protocols. Thirty-two channels may be dedicated to protocol A, 64 channels may be dedicated to each of protocols B and C, and so on. Thus, the load monitor determines the number of channels to devote to each of the protocols.

Each channel maps to a set of PLD resources. In one embodiment, not all resources are initially allocated. For example, 10% of the total channels are not allocated to any protocol and are held as a reserve of resources for peak activity levels. At step 154, the configuration control is signaled to begin the initial configuration. The load monitor provides the configuration control with information identifying which PLD resources are to be configured with which configuration bitstreams. The manner in which PLD resources and configuration bitstreams are identified will depend on the specific functions and the capabilities of the particular PLD. Furthermore, if a single PLD supports more than one function, then a number of configuration bitstreams may be required to cover the possible combinations for allocating resources of the PLD between the different functions.

Once the PLDs have been configured and the system is operational, at step 156, the load monitor establishes communications with the functions on the PLDs. At step 158, the load monitor begins monitoring the activity levels of the functions. The load monitor samples (step 160) the activity levels every *t* interval of time for each function. A typical measure of activity level for a communications

protocol channel is the number of packets processed per second. If a protocol is supported across multiple channels, the activity level is the total activity across all the channels implementing the protocol.

Decision step 162 tests whether the activity level of any of the functions is decreasing. When a function shows a decreasing activity level over *n* samples and the overall decrease is greater than a given threshold, some of the resources allocated to the function are freed for re-allocation to other applications. The value of *n* and the threshold level are chosen so that reallocation is not performed in response to random and instantaneous changes. Each function may have a different threshold that varies based on the number of resources allocated to the function since functions having greater numbers of resources will be able to accommodate greater activity levels. In various embodiments, the functions implemented on the PLDs are programmed to periodically report activity levels to the load monitor either via an on-board PLD register or by transmission to the load monitor.

Step 164 determines how resources are to be reallocated between the functions implemented on the PLDs (FIG. 3A describes how the reallocated resource levels are determined for decreasing activity levels). At step 166, the load monitor signals the PLD(s) that will be reconfigured to complete work-in-process prior to reconfiguration. The work may be completed by virtue of finishing the work or transferring the work to a different resource (e.g., another PLD that will not be reconfigured). For I/O tasks on resources to be freed, a number of steps are performed:

1. The output ports from the resources to be freed are connected to output ports of another resource that implements the same function. For example, if channels A and B both implement protocol 1 and the resources from channel A are to be freed, the output ports of channel A are connected to unassigned output ports of channel B, or if time-slicing is used, to outputs of channel B having unused capacity.
2. Direct input data from the resource to be freed to another resource that implements the same function. For example, (continuing the example of step 1) the input data directed at channel A is redirected to the input ports of channel B.
3. Disable the output ports of the resource to be freed.
4. Disable the input ports of the resource to be freed.

The connection and redirection of inputs and outputs can best be facilitated by either special I/O circuitry (for example, a crossbar switch), or a PLD reconfigured to re-map input and output connections.

At step 168, the configuration control element is signaled to begin the reconfiguration. The PLD resources that are reconfigured are those identified at step 164.

If decision step 162 finds that none of the functions have decreasing activity levels, the process is directed to decision step 172 to test whether any of the functions have increasing activity levels. If there are no functions having decreasing or increasing activity levels, the process returns to sampling the activity levels at step 160.

If any of the functions are detected as having increased activity levels, the process is directed to step 174. When a function shows an increasing activity level over *n* samples and the overall increase is greater than a given threshold, resources from the reserve a allocated to the function(s). The values of *n* and the threshold level are chosen so that reallocation is not performed in response to random and instantaneous changes. The thresholds associated with increasing activity levels may be different from the thresholds associated with decreasing activity levels, and each function may have a different threshold that varies based on the number of resources allocated to the function since functions having greater numbers of resources will be able to accommodate greater activity levels.

At step 174, it is determined how resources from the reserve are to be allocated to the function(s) (FIG. 3B describes how the reallocated resource levels are determined for increasing activity levels). The process then continues at step 168 as described above.

FIG. 3A is a flowchart of a process for determining the levels at which PLD resources should be reallocated between a number of functions when the activity level of a function is decreasing. The number of resources to free from a function is determined at step 202 and is based on the following model. Each resource can service a maximum load of R_L for its given function. Thus, if M resources are assigned to a function, then the maximum sustainable activity level is $M*R_L$. If the load monitor determines that the activity level has decreased such that a current activity level = $(M-P)* R_L$ $(P<=M)$, then the $(P-1)$ resources

are freed (one is left in reserve with the function). At step 204, the number of resources of the function is reduced by the amount determined at step 202. For example, the number of channels that implement a particular protocol is reduced.

Decision step 206 determines whether the freed resources should be allocated to other functions or held in reserve. The decision is made based on a projection of future activity levels of the other functions. The projection can be linear, least squares, or another technique that best models the overall system behavior. In one embodiment, the projection is made based on past behavior. Sample activity levels are taken by the load monitor. The values associated with the activity levels are stored over a sufficiently long window of time. The length of the window is based on the usage environment. Past performance is modeled statistically to predict future behavior. If the predicted future behavior is either a decrease or increase in the activity level, then reconfiguration and re-assignment may be initiated.

At step 208, the allocation of freed resources to functions is determined based on the projected, proportional needs of the functions. For example, if the activity level of a first function is projected to increase by 30% and the activity level of a second function is projected to increase by 10%, then the ratio of the projected increases is used when making the allocation. Assuming both the first and second functions begin with the same initial resource allocation, then the first function will be allocated three-fourths of the resources and the second function is allocated one-fourth of the resources.

If none of the activity levels is increasing, decision step 206 directs control to step 210, and the freed resources are added to the reserve of free resources. Parameters that describe the resources to allocate to the different functions are then returned to the process of FIG. 2.

FIG. 3B is a flowchart of a process for determining the levels at which PLD resources should be allocated from the reserve when the activity level of a function is increasing. At step 222, the number of resources to remove from the reserve and allocate to the one or more functions is determined.

The allocation of resources takes into account the increasing needs of the entire system (i.e., across multiple functions). The allocation process also ensures that the entire reserve is not depleted so that future increased activity levels can be handled. In one embodiment, the following model is used to manage the reserve of resources:

1. The load monitor tracks the total percentage of resources allocated to each function and the percentage of resources presently in the reserve. The total of all allocated resources is 100% ($R_1+R_2+R_3+R_4+\ldots+R_{RESERVE}$ = 100%, where R_X is the percent resources allocated to function *X*).
2. The load monitor tracks the predicted increase (or decrease) of resources for each function. Each predicted increase (or decrease) is represented as a percentage of the current resources allocated to a function.

When allocating from the reserve, a percentage of the reserve is maintained for future allocations. The portion of the reserve maintained for future allocations can be established as a fixed percentage or a sliding percentage that varies as a function of the activity levels of all the functions, for example, a smaller percentage when the functions are more active and a larger percentage when the functions are less active. The resources in the reserve will slowly decline to zero as the system reaches its saturation capacity. When the reserve is empty, only decreases in the activity levels of functions can trigger reallocations.

Resources are proportionally allocated from the reserve based on the projected increased activity levels of the functions. In addition, functions having the greatest projected increase are considered first. For example, if function A's activity level is projected to increase 30%, function B's 20%, function C's 10%, and 50 units of the reserve are eligible to allocate, then function A should maximally get 25 units (1/2 * 50 units), function B should maximally get 17 units (1/3 * 50), and function C should maximally get 8 units (1/6 * 50).

After the maximum allocations have been established, the allocations and assignment priority are evaluated against the actual predicted unit increase. For example, if function A is projected to have a 30% increase but currently has 10 resource units, then an allocation of 3 units rather than 25 would suffice. Thus, function A is allocated the 3 units and the difference is left in the reserve and eligible for allocation. This allows allocation to function B to be as much as needed.

The proportional allocation is then redone as between functions B and C using the eligible 47 units (50 - 3). Since function B's projected increase was 20% and function C's projected increase was 10%, function B may be proportionally allocated 32 units (~(2/3 * 47)), and function C may be allocated 15 units (47 - 32). However, if function B is currently allocated 400 units, then a 20% projected increase implies that an additional 80 units may be required (20% * 400). However our rule allows only 32 to be allocated. We note however that more were actually requested and that as many as an additional (80-32) or 48 could be used. If C is currently allocated 10 units, then a projected 10% increase implies the allocation of 1 unit (10% * 10). This means that now (47-32-1) or 14 units remain.

If any of the functions were allocated fewer resources than called for by the projection of increased activity level (for example, function B) and not all of the eligible resources were allocated in the first pass, then the process described above is repeated for those functions having projected activity levels that exceed the resources allocated in the first pass. In the present example only function B has an outstanding request for 48 units, and 14 units remain eligible for allocation. Thus, the 14 units are allocated to function B, leaving a deficiency of 34 units (80 - (32 + 14)).

It will be appreciated that the inability to allocate sufficient resources to meet the projected need may or may not impact performance levels of the functions. If the projected increased activity levels are not reached, there is no impact on performance levels of the functions. If the projections were accurate and a function was not

allocated all the resources necessary to meet the projections, then the performance level of the functions may be impaired.

After the selected resources of the PLD have been reconfigured in accordance with the selected allocation of resources, the load monitor continues to sample activity.

At step 224, the number of resources maintained in the reserve is reduced by the amount determined at step 222. For example, the number of channels that implement a particular protocol is reduced.

At step 228, the allocation of resources removed from the reserve amongst one or more functions is determined based on the projected, proportional activity levels of the functions. The projected activity levels and proportional allocation are determined as described above in association with FIG. 3A. Parameters that describe the resources to allocate to the different functions are then returned to the process of FIG. 2.

FIG. 4 is a flowchart of a process for controlling reconfiguration of PLDs in response to control parameters from the load monitor.

At step 252, one or more configuration bitstreams are selected in response to the signals from the load monitor. The particular configuration bitstreams that are selected will depend on the PLD resources to be reconfigured and the selected functions.

At step 254, the PLD resources that are to be reconfigured are selected.

The selected configuration bitstreams are then downloaded to the PLDs at step 256.

At step 258, the reconfigured PLDs are restarted with the reprogrammed functions.

Finally, at step 260 the load monitor is signaled that the reconfiguration is complete.

The present invention is believed to be applicable to a variety of systems having a plurality of PLD-implemented functions. Other aspects and embodiments of the present invention will be apparent to those skilled in the art from consideration of the specification and practice of the invention disclosed herein. It is intended that the specification and illustrated embodiments be considered as examples only, with a true scope and spirit of the invention being indicated by the following claims.